

Estudi i implementació de monitoratge web per processos a Prepersa

Alex Miralles Torrente

Resum—Les grans empreses avui dia gestionen grans quantitats de dades generades per tot tipus de transaccions informàtiques i l'estat d'aquestes dades és rellevant per tal de conèixer les condicions dels diferents fronts de les empreses. En aquest paper s'exposa l'estudi, el disseny i la implementació d'un sistema que permet accedir a temps real a les dades clau que proveeixen els processos de Prepersa, empresa que forma part de Catalana Occident. A partir de l'estudi i disseny de diverses propostes que resolen aquesta necessitat, se selecciona una proposta i s'arriba a una solució mitjançant una aplicació web que a partir de la base de dades de Prepersa ofereix informació de control a l'usuari i l'accés a un aplicatiu basat en Kibana que interpreta els logs i els mostra en una interfície web en forma de gràfics i mètriques.

Paraules clau— Backend, Frontend, Dashboard, ELK, DII, Kibana, procés

Abstract—Large companies today manage large amounts of data generated by all types of computer transactions and the status of this data is relevant in order to know the conditions of the different fronts of companies. This paper sets out the study, design and implementation of a system that allows real-time access to the key data provided by the processes of Prepersa, a company that is part of Catalana Occidente. From the study and design of various proposals that solve this need, a proposal is selected. A solution is reached through a web application that provides control information to the user from database Prepersa and access to a Kibana-based application that interprets the logs and displays them in a web interface in the form of graphs and metrics.

Index Terms— Backend, Frontend, Dashboard, ELK, DII, Kibana, procés

1 INTRODUCCIÓ

A VUI dia les empreses generen una enorme quantitat de dades que permeten portar un control de les seves transaccions i processos. Aquestes dades són utilitzades entre altres motius per tenir un control de l'estat de l'empresa comportant-se com indicadors que defineixen si està havent irregularitats en el comportament del sistema. Una de les principals preocupacions de la majoria d'aquestes empreses és el tractament d'aquestes dades per portar un control continuat dels indicadors anteriorment mencionats per tal de realitzar accions preventives o reactives en vers a potencials amenaces o problemes.

Aquesta situació té lloc en el departament d'informàtica de Prepersa, empresa que forma part del grup Catalana Occident [1]. En Prepersa es duen a terme al voltant de 2000 execucions de processos diàriament i cada execució genera dades en logs i en la base de dades. Existeix la necessitat de tractar i estructurar aquestes dades per tal de mostrar-les de forma que es puguin interpretar a temps real de manera senzilla, veient ràpidament quin és l'estat general dels processos de l'empresa i també l'estat específic de cada procés. Si solventa aquesta necessitat, llavors permetrà a l'equip identificar errors, trobar solucions a

aquests errors i anticipar i prevenir nous errors.

En aquest document s'exposarà la solució a la qual s'ha arribat per tal de resoldre la necessitat de controlar la informació dels processos de l'empresa Prepersa. Primer es realitzarà una introducció en la secció 1. Per tal de contextualitzar es farà una descripció del sistema de l'empresa en la secció 2. Posteriorment es mencionaran els objectius a assolir del projecte en la secció 3. En la secció 4 es parlarà de la metodologia emprada per la realització del projecte. S'introduirà la recollida de requeriments en la secció 5. En la secció 6 es realitzarà un estudi de l'estructura de les possibles solucions al problema, en la secció 7 es veurà com s'ha elaborat el disseny de la proposta seleccionada, en la secció 8 es trobarà el desenvolupament i implementació de la proposta i en la secció 9 es descriuran les proves realitzades per validar la solució. Posteriorment es parlaran dels resultats obtinguts amb l'aplicació en la secció 10, expressant si ha sigut satisfactòria o no. També s'explicaran a mode de resum del document les conclusions del projecte en la secció 11, apuntant els punts més rellevants del mateix.

2 CONTEXT

En aquest apartat es descriurà el sistema de l'empresa i els principals elements del sistema base que utilitzarà l'eina de monitoratge. Primerament es descriurà el sistema de l'empresa, després s'esmentarà el model de dades emprat basat en logs i el funcionament de l'eina per plasmar gràficament

- Alex Miralles Torrente, estudiant d'Enginyeria informàtica en menció de Tecnologies de la Informació a la universitat Autònoma de Barcelona.
E-mail: alex.mirallest@e-campus.uab.cat.
- Victor García Font, tutor Departament de la Informació i de les Comunicacions a la Universitat Autònoma de Barcelona.
E-mail: Victor.Garcia@uab.cat.

les dades extretes del model de dades.

2.1 Sistema de l'empresa

Prepersa es dedica a desenvolupar el portal web dedicat que utilitzen els pèrits de Catalana Occident per gestionar els seus encàrrecs, dades, pagaments, baixes, etc.

L'empresa disposa de diversos servidors dedicats que allotgen els Backend, Frontend i Processos de Prepersa. També disposa d'una base de dades i una rèplica del portal web per cada entorn de treball, en el cas de Prepersa hi trobem l'entorn de desenvolupament, l'entorn de preproducció i el de producció.

El funcionament del portal web està basat en un conjunt de Backend i Frontends. Els Backend s'encarreguen d'extreure i inserir informació a la base de dades i els Frontend s'ocupen de mostrar les dades extretes pels Backend en un format que l'usuari final pugui utilitzar-les i també permet a l'usuari interactuar amb el sistema per introduir-hi noves dades.

Els processos juguen un paper molt important en Prepersa tot i que actuen amb el portal web de forma més indirecta que els Frontend i Backend. Els processos són els encarregats de tractar dades que poden provindre d'altres companyies, la base de dades de Prepersa, fitxers... per elaborar un producte final que si que podrà ser útil per al portal web o per afers interns de la mateixa empresa. Un exemple podria ser un procés dedicat a la migració de dades d'un sharepoint extern a la base de dades de Prepersa o bé, un procés que executi els scripts necessaris per passar Backend i Frontend cap als diferents entorns de treball abans mencionats.

L'activitat tant dels Backend, Frontend i processos és enregistrada en els seus respectius servidors en logs en format de .txt. Aquest enregistrament es realitza per tenir un control de qualsevol dels esdeveniments que es poden produir dins del sistema de l'empresa, així quan succeeixi algun problema es podrà buscar les causes en els logs.

2.2 Model de dades

El principal model de dades que s'utilitzarà per al monitoratge consisteix en una sèrie de logs que es van informant amb l'execució de cada procés. Els logs són arxius en format ".txt" que es van creant de forma diària, de manera que es manté un històric que està compost per un log per dia enregistrat.

També s'utilitzarà un model de dades basat en Database que s'inclourà en la secció de desenvolupament, ja que no ha vingut donat pel sistema actual de l'empresa i s'ha hagut de dissenyar i desenvolupar des de zero.

En les següents seccions s'explica en detall el funcionament i estructura dels logs.

2.2.1 Generació dels logs

Entrant més en detall de com es generen els logs podem dir que cada procés creat ha de seguir una certa estructura plantejada per l'empresa que facilita la recopilació de la informació per als logs, enviament de mails, control d'errors, etc.

El tret principal de l'estructura és que s'hi ha de referenciar una llibreria dll anomenada Base.dll. Entre altres coses aquesta llibreria consisteix en codi que s'executa a l'inici de

l'execució de qualsevol procés per crear informació sobre l'instància utilitzada, l'hora de l'inici de l'execució, nom del procés; i al final de l'execució d'on extreu la informació afegida pel mateix programador, els errors generats durant l'execució i el resultat d'aquesta.

La llibreria Base.dll fa una crida al logger que escriurà utilitzant l'estructura mencionada en l'anterior apartat en les rutes definides.

2.2.1 Estructura dels logs

Tots els logs es generen de forma automàtica a partir d'un procés comú a tots ells anomenat procés Base. Aquest procés recopila les dades disponibles en els processos i les crida als BE utilitzats per aquests i les escriu en diversos logs. Hi ha 3 tipus de logs:

Error.log: Conté únicament informació dels processos que han generat un error de qualsevol mena. Concretament genera una línia de text per cada error generat. Camps de dades que conté:

- Data i hora exacta
- Nom del procés
- Instància (identificador únic per cada instància)
- Missatge d'error

Resultado.log: Aquí s'enregistren dades de totes les execucions de processos realitzades, tant si són errònies com si són correctes. En aquest cas es genera una sola línia de text per cada execució. Camps de dades que conté:

- Data i hora exacta
- Nom del procés
- Instància (identificador únic per cada instància)
- Codi del resultat

Traza.log: Cada procés genera individualment el seu propi Traza.log el que enregistrarà informació de tot tipus com quan inicia el procés i quan acaba, control d'errors, informació sobre les accions emprades, informació dels backend que utilitza, etc. Aquest log pot generar moltes línies de text per cada execució, ja que dependrà del programador del procés. Els camps de dades que conté són els següents:

- Data i hora exacta
- Nom del procés
- Instància (identificador únic per cada instància)
- Tipus: Indica de quin tipus de missatge és la línia en el que es troba (Error, Informació, Inici de procés, Finalització de procés o Resultat de l'execució)
- Missatge: Aquí és on es descriu l'esdeveniment que es tracta en la línia en qüestió.

2.3 ELK

ELK es una eina utilitzada per Catalana Occident per monitorar principalment l'estat dels backends que estan funcionant en producció. Aquesta eina està composta per 3 projectes que s'encarreguen de treballar en conjunt per a emmagatzemar, unificar, cercar i visualitzar dades de forma eficient. Els projectes són els següents:

1. Logstash:

És una tecnologia open source basada en jRuby capaç de recollir dades en temps real. Logstash unifica i norma-

litz grans volums de dades que poden tenir orígens diferents, permeten així realitzar cerques o visualitzar dades [4].

2. Elastic Search:

És un motor de cerca i analítics open source basat en Lucene, utilitza una interfície RESTful y exporta les seves dades mitjançant documents JSON. Aquesta eina és el nucli principal d'Elastic-Stack, ja que centralitza totes les dades introduïdes per Logstash, permet realitzar cerques de gran velocitat i es molt escalable degut, ja que és un sistema distribuït [5].

3. Kibana:

És l'eina encarregada de rebre les dades provinents de Elastic Search i representar-les en forma de diagrames, taules, mètriques o cerques. L'enfocament de la representació d'aquestes dades es basa a tractar-les de forma dinàmica, actualitzant constantment les dades mostrades en els diferents diagrames i permetent utilitzar tot mena de filtres, fent-lo ideal per a al monitoratge a temps real de dades.

Com es veurà més endavant, ELK és la tecnologia emprada per arribar a l'objectiu principal del projecte, el monitoratge dels processos a temps real i de forma senzilla.

3 OBJECTIUS

L'objectiu d'aquest projecte és aconseguir mostrar a temps real l'estat dels processos i facilitar les accions a emprendre pel departament informàtic de Prepersa en casos d'errors o prevenció de problemes. Els següents objectius permetran arribar satisfactòriament a complir l'objectiu principal:

- Recollir els requeriments mitjançant entrevistes amb els clients utilitzant dinàmiques de grup.
- Creació d'un backend per cada funcionalitat del dashboard web inicial.
- Creació d'un model de dades que assumeixi totes les dades necessàries per al dashboard i tingui una estructura reutilitzable.
- Millorar les funcions dels processos com la reducció d'emails enviats o la detecció de processos interromputs.
- Crear una eina adaptable que admeti modificacions en la seva estructura i es puguin afegir noves funcionalitats i nous processos sense dificultats.
- Eina intuïtiva per als clients, s'avaluarà mitjançant recorreguts guiats per l'aplicació i tests de satisfacció als clients.
- Tenir documentació per cada fase del projecte.

4 METODOLOGIA

La metodologia emprada en aquest projecte es Scrum, metodologia basada en Agile que permet abordar problemes complexos a causa dels seus processos iteratius i incrementals es poden desenvolupar projectes en els quals els requeriments van variant en el temps [2].

En aquest projecte s'han realitzat iteracions curtes de dues setmanes, els processos són les diferents reunions necessàries per portar el control del desenvolupament i la metodologia. Els processos són els següents:

1. **Sprint Planning:** Fase inicial de la iteració se seleccionen els requeriments a desenvolupar.

2. **Sprint:** Fase de desenvolupament dels requeriments. Cada dia es realitza un *Daily sprint Meeting*, en el que es posa al dia de l'avanç de cada individu de l'equip.

3. **Sprint Review:** Es defineix el següent sprint, passant la feina no feta al product backlog.

4. **Sprint Retrospective:** Reunió orientada a analitzar l'últim sprint per tal de treure conclusions útils per millorar com a equip.

En la figura 1 es pot veure els processos de cada iteració Scrum.

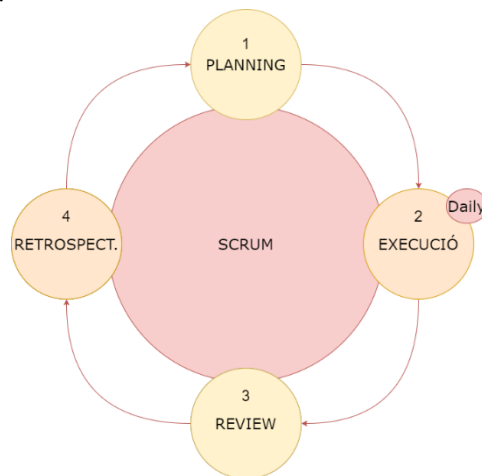


Fig. 1. Processos Scrum.

S'ha escollit aquesta metodologia perquè tot i que l'objectiu principal del projecte era molt clar des de el principi, les eines a emprar i el desenvolupament no estava ben definit. Scrum permet a través dels seus processos adaptar el projecte a les necessitats canviants dels clients.

5 RECOLLIDA DE REQUERIMENTS

A l'inici del projecte es van realitzar diverses reunions amb els clients per tal de definir quines característiques havia de complir l'aplicació de monitoratge que volien que es desenvolupés.

La principal tècnica d'obtenció de requeriments utilitzada va ser el brainstorming que consisteix en una dinàmica de grup en la que tots els participants aporten idees, aquestes s'anoten i posteriorment es realitza una anàlisi de les idees generades. Aquesta tècnica ha sigut molt útil, ja que els requeriments de l'aplicació no estaven ben definits i es van poder generar moltes idees a partir de la creativitat dels participants de forma organitzada.

Alguns exemples dels requeriments més rellevants obtinguts són:

Requeriments Funcionals:

- L'usuari ha de poder veure i especificar quins processos vol monitorar.
- L'aplicació ha de mostrar d'actualitzar automàticament la informació mostrada.
- L'usuari ha de poder visualitzar l'estat de tots els tipus de processos (GCO, Prepersa, Aemet, App...)

- L'aplicació ha de ser accessible a través d'una pàgina web.

Requeriments no funcionals

- L'aplicació ha de mantenir com a mínim 30 dies de registres de monitoratge.
- La freqüència d'actualització del monitoratge no pot ser major de 10 minuts.
- La disponibilitat de l'eina ha de ser com a mínim del 99%.
- El temps de resposta no pot superar els 5 segons.

6 ANÀLISI

Un cop obtinguts els requeriments es va passar a la fase d'anàlisi del sistema de l'empresa per conèixer les possibilitats a l'hora de dissenyar i proposar una solució que satisfés aquests requeriments. Les propostes resultants van ser les següents.

6.1 Monitoratge utilitzant la base de dades

La primera idea que es va plantejar va ser desenvolupar el monitoratge utilitzant la base de dades de l'empresa com a magatzem de les dades històriques del monitoratge. Aquesta idea consistia a realitzar insercions a la base de dades de Prepresa en cada execució de processos, això permet tenir molta flexibilitat a l'hora d'extreure qualsevol mena d'informació dels processos. Un cop inserides les dades a partir d'un dashboard web desenvolupat amb el sistema clàssic de backends i frontends s'obtindria la informació de la base de dades de forma periòdica. En aquesta aplicació web es trobarien funcionalitats com inserció, edició, eliminació de processos, així com informació dels mateixos en format de taules.

Aquesta solució implicava una gran càrrega de treball sobre la base de dades, ja que es desitja desenvolupar una aplicació capaç de donar informació de l'estat dels processos en temps real i això implica realitzar un gran nombre d'operacions SQL contínuament. Per tant, per tenir una aplicació que no afecti al rendiment amb aquest sistema s'hauria de reduir la freqüència d'actualització del monitoratge i també hi hauria una gran limitació en la informació històrica que es podria acumular en la base de dades, ja que s'executen més de 2.000 processos cada dia i ràpidament es generaria una gran quantitat de registres.

6.2 Monitoratge utilitzant logs amb Kibana

Durant l'anàlisi de la primera proposta va aparèixer la possibilitat de treballar amb l'eina ELK, ja que el grup Catalana Occident, va disposar a les empreses pertanyents al mateix d'accés a aquesta eina.

Aquesta proposta consisteix en dos punts principals, el monitoratge dels processos a través de l'aplicació Kibana i un dashboard basat en GAAN, GAAN és el framework utilitzat per desenvolupar Frontends que s'utilitza en Catalana Occident, on es podrà gestionar informació general de cada procés.

L'aplicació basada en GAAN que es planteja consisteix en una taula d'informació extensa en la qual es podran veure tot mena de dades útils que permetran treballar amb més eficiència amb els processos que calgui. La informació a mostrar és: nom del procés, descripció, rutes en els dife-

rents entorns (desenvolupament, preproducció, producció), rutes dels logs, freqüència d'execució, dependències a nivell BackEnd, importància del procés, enllaç a Kibana amb el filtre corresponent al mateix procés, etc. Aquesta pantalla permet a l'usuari afegir, eliminar, editar o veure en detall els diferents processos mitjançant uns botons que naveguen a un pop-up, de forma similar a l'anterior proposta.

El monitoratge es basa en la introducció de les dades que generen els processos de l'empresa en els logs propis cap a Elastic Search mitjançant un parseig de dades entre els logs i els servidors d'elastic search. Un cop introduïdes les dades en Elastic Search s'utilitzaran els índexs generats per extreure la informació i utilitzar-la en forma de gràfics i estadístiques que donaran una visió clara de l'estat dels processos. En aquesta aplicació es disposa d'un motor de cerca molt potent que es capaç de llegir ràpidament dins de les dades de logs introduïdes per buscar i filtrar els resultats que es mostraran en les gràfiques, permetent així discriminar quins processos es volen mostrar de forma intuïtiva. Aquesta eina permet actualitzar les dades del dashboard segon a segon i no suposa cap càrrega cap als servidors propis de l'empresa, ja que és un servei extern.

6.3 Decisió final

Un cop definides les propostes es va realitzar una valoració tenint en compte els pros i contres de cada opció per prendre la proposta més profitosa. En la següent taula es poden observar els principals factors que es van tenir en compte en aquesta decisió.

	Base de dades	Kibana
Pros	-Control total de l'app. -Major flexibilitat en les funcionalitats.	-Baix cost de manteniment. -Implementació simple. -Alta disponibilitat. -Actualització de dades amb freqüència d'1 segon. -Alliberament de càrrega a la base de dades de Prepresa.
Contres	-Sobrecàrrega a la base de dades de Prepresa. -Actualització de dades amb període superior a 1 minut. -Manteniment costós. -Absència d'elements de disseny gràfics.	-Dependència externa. -Dades de monitoratge limitades als logs de processos. -Funcionalitats definides per Kibana.

Es va decidir prendre la proposta de realitzar el monitoratge a través de l'eina Kibana com a la definitiva perquè tot i tenir menys informació disponible dels processos respecte a l'anterior, permet tenir un monitoratge sense delay i no comporta problemes de sobrecàrrega en la base de dades de l'empresa.

7 DISSENY

7.1 Model de dades

Per desenvolupar el dashboard inicial d'informació general de processos es necessari tenir una base de dades que respaldi l'aplicació.

El model de dades consisteix en un model relacional que es caracteritza per l'ús de les relacions. Aquestes relacions es poden entendre com tuples que connecten dues taules associant-les entre elles. Alguns dels avantatges d'aquest model és que permet tractar dades dinàmicament, evita les duplicitats de registres i proveeix d'integritat referencial. Per aquest projecte es molt adient, ja que diverses taules del model podrien generar duplicats si no tractesim amb aquest tipus de model.

En la figura 2 podem veure el model amb les diferents taules relacionades amb colors.

Proceso	Tipo	Backend	Tipo
IdProceso	int, incr, pk	IdBackend	int, incr, pk
NonProceso	varchar(20)	NonBackend	varchar(20)
DesProceso	nullable	RutaTFSBackend	nullable, varchar(100)
IdProcesoPeriodicidadTipo	int, fk	TimCreacion	datetime
IdProcesoImportanciaTipo	int, fk	UtdCreacion	varchar(70)
RutaTFSProceso	nullable, varchar(100)	IndBaja	bit
RutaTFSBackend	nullable, varchar(100)	TimModificacion	datetime
NonTareaProceso	nullable, varchar(20)	UtdModificacion	varchar(70)
TimCreacion	datetime		
UtdCreacion	varchar(70)		
IndBaja	bit		
TimModificacion	datetime		
UtdModificacion	varchar(70)		

ProcesoPeriodicidadTipo	Tipo	ProcesoImportanciaTipo	Tipo
IdProcesoPeriodicidadTipo	int, incr, pk	IdProcesoImportanciaTipo	int, pk
DesProcesoPeriodicidadTipo	varchar(50)	DesProcesoImportanciaTipo	varchar(20)
NumProcesoPeriodicidadTipo	int	TimCreacion	datetime
IdProcesoIntervaloTipo	int, fk	UtdCreacion	varchar(70)
TimCreacion	datetime	IndBaja	bit
UtdCreacion	varchar(70)	TimModificacion	datetime
IndBaja	bit	UtdModificacion	varchar(70)
TimModificacion	datetime		
UtdModificacion	varchar(70)		

ProcesoIntervaloTipo	Tipo	ProcesoBackendRel	Tipo
IdProcesoIntervaloTipo	int, incr, pk	IdProcesoBackendRel	int, incr, pk
DesProcesoIntervaloTipo	varchar(20)	IdProceso	int, fk
TimCreacion	datetime	IdBackend	int, fk
UtdCreacion	varchar(70)	TimCreacion	datetime
IndBaja	bit	UtdCreacion	varchar(70)
TimModificacion	datetime	IndBaja	bit
UtdModificacion	varchar(70)	TimModificacion	datetime
		UtdModificacion	varchar(70)

Fig. 2. Model de dades.

7.2 Prototips

La creació de prototips en el disseny serveix per plasma la idea a la qual es vol arribar i servir com a guia per als desenvolupadors que coneixeran millor els detalls del projecte i també per presentar als clients i verificar si el disseny del prototip s'acosta a la solució que busca resoldre els seus problemes.

En aquest cas s'ha decidit realitzar un prototip web basat en Marvel App amb el que els usuaris podran navegar pel prototip com si es tractés de l'aplicació. Això ens permet verificar la validesa del disseny del projecte, comprovar que les funcionalitats són les desitjades i finalment ajudar als desenvolupadors a tenir un camí clar en el desenvolupament de l'aplicació.

Per dissenyar el prototip s'han utilitzat les imatges que es poden veure en la figura 3 on també es poden apreciar les relacions entre elles.

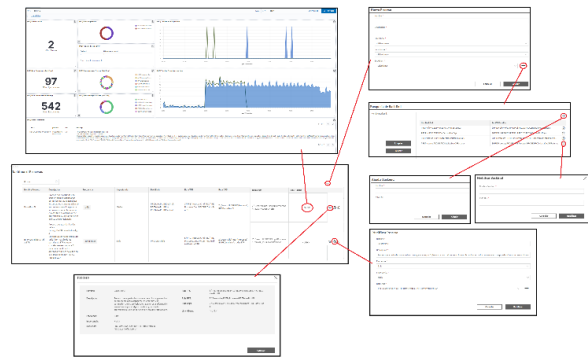


Fig. 3. Pantalles del prototip.

7.3 Casos d'ús

D'acord amb l'elicitació de requeriments i les diverses reunions i processos de Scrum en els que s'ha contactat amb els clients s'hi han debatut quines serien les diverses accions que podria realitzar un usuari en l'aplicació. A l'hora de desenvolupar s'han d'establir els mínims requerits per a que l'aplicació sigui el producte que es busca i per tenir clares aquestes funcions s'ha generat un diagrama de casos d'ús.

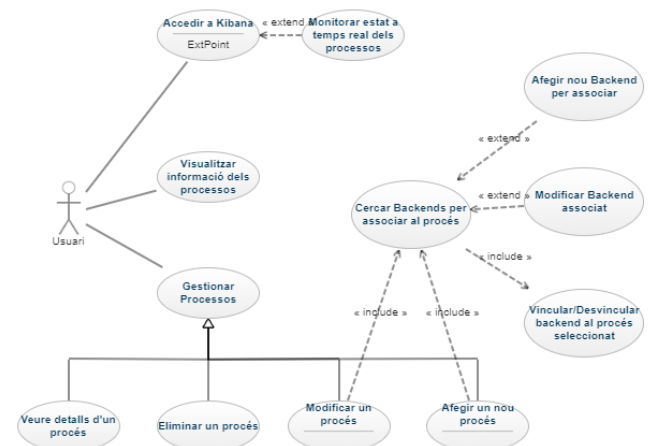


Fig. 4. Diagrama de casos d'ús.

En la figura 4 es poden veure les diferents accions que pot realitzar l'usuari junt amb les seves relacions amb la resta d'accions.

8 DESENVOLUPAMENT

En aquesta secció es descriurà com s'han implementat els dos dashboard a l'aplicació, incloent-hi la creació de la base de dades, codi desenvolupat, introducció de dades als servidors d'Elastic Search i la creació de gràfics i mètriques amb Kibana.

8.1 Dashboard Inicial

Primerament s'explicarà la creació del model de base de dades que utilitzara per a aquest dashboard. Després es descriurà com s'han creat els Backend necessaris per extreure i inserir la informació de la base de dades (DB) i finalment es parlarà de la creació dels Frontend amb el framework proporcionat per Prepresa anomenat GAAN.

8.2 Dashboard Monitoratge

Per desenvolupar el dashboard encarregat del monitoratge a temps real de processos primer s'ha hagut de realitzar una introducció de les dades dels logs de processos de Prepersa als servidors de Elastic Search i després s'ha creat el dashboard en l'aplicació de visualització gràfica Kibana. En les seccions a continuació es detalla el procés.

8.2.1 Introducció de logs a Elastic Search

Aquesta secció descriu com s’han introduït les dades al sistema de ELK per a la seva indexació i posterior ús a Kibana, que permetrà utilitzar les dades de forma analítica i gràfica. Per tal d’introduir les dades dels logs al servidor d’Elastic Search inicialment s’han creat tres pipelines, que són els encarregats de donar informació al servidor d’Elastic Search de les rutes dels logs que volem monitorar. Així doncs, es va decidir prescindir de Logstash degut a que les dades que tenim en els logs són homogènies i no precisen d’una unificació perquè les dades siguin consistents entre sí. El servidor executa contínuament un procés anomenat filebeat que s’encarrega d’enviar la informació dels pipeline als ingest node que són els encarregats de parsejar les dades introduïdes tenint en compte el seu format [6], que l’indiquem a través de fitxers JSON, la figura 9 és un exemple clar.

[illegible]

Fig. 9. Fitxer JSON amb patrons Grok.

Els ingest node adapten les dades que els proporcionem de forma que Elastic Search i Kibana puguin interpretar-les i també definim l'estructura que tindran utilitzant patrons grok [7]. En aquest cas el primer grok donarà processa la ruta i nom del fitxer i el segon grok processa el contingut dels fitxers que consisteix en els diferents camps dels logs: Data, Nom del procés, instància, etc.

Tenint configurada l'estructura perquè Elastic Search pugui extreure la informació dels logs, des de l'eina Kibana se'ns disposarà de tres fonts de dades per poder representar-la a l'hora de crear les visualitzacions dels dashboards, una per cada tipus de log.

8.2.2 Creació dashboard a Kibana

Un cop introduïts els logs als servidors de Elastic search les dades dels logs de Prepersa estan formatejades per tal de ser interpretades per l'aplicació Kibana.

El primer que s'ha realitzat es la creació del dashboard de Kibana en blanc que contindrà totes les visualitzacions. Un cop creat s'aniran fent una a una diferents visualitzacions. El procés de creació de visualitzacions que s'ha seguit és el següent:

1. **Escollir tipus de visualització:** Aquí s'hi han se-

leccionat totes les plantilles gràfiques que s'utilitzen per representar les dades. En aquest projecte s'han utilitzat mètriques, taules de dades, timelines i diagrames de seccions.

2. **Indicar dades a mostrar:** En aquest procés s'han seleccionat els filebeats dels que s'extreu la informació, en aquest cas son 3 filebeats corresponents a cada tipus de log de processos (resultats, errors i traces).
3. **Selecció de paràmetres dels gràfics:** Aquí s'hi han inclòs les característiques de cada gràfic com: que es vol mostrar, marges temporals, detalls gràfics, opcions a mostrar, etc.
4. **Addició i adaptació al dashboard:** Un cop creada la visualització s'ha afegit al dashboard, on s'ha ubicat i adaptat cada visualització de forma que encaixi amb el model desitjat.

Un cop creat el dashboard amb les visualitzacions s'ha decidit utilitzar una freqüència d'actualització de 5 segons de les dades, el marge temporal de les dades seleccionat és del dia en el qual que ens trobem.

Sabent el procés de creació a continuació s'explica quines visualitzacions s'han implementat en el dashboard i el motiu per el qual s'ha fet.

- **Errors totals:** Comptador que indica quants errors s'han generat. Útil per saber si el sistema està funcionant correctament o no.
- **Execucions totals:** Comptador que indica quants processos s'han executat. Útil per contrastar amb el comptador d'errors.
- **Diagrama d'errors per procés:** Diagrama de seccions que indica quants errors ha generat cada procés en error. Útil per conèixer quins processos fallen i la magnitud del problema.
- **Diagrama d'execucions per procés:** Diagrama de seccions que indica el nombre d'execucions que ha fet cada procés. Útil en contrast al diagrama anterior.
- **Timeline amb nombre d'errors:** Diagrama de línies que indica la quantitat d'errors en marges de 10 minuts. Útil per programar l'hora d'execució dels processos al conèixer millor les concurrències d'execucions problemàtiques o resetejos de servidors de processos.
- **Timeline amb nombre d'execucions:** Diagrama de línies que indica la quantitat d'execucions en marges de 10 minuts. Útil com a indicador que els servidors estiguin caiguts o de sobrecàrregues inesperades en els mateixos.
- **Taula amb missatges d'errors:** Taula que conté el nom del procés codi d'error i missatge d'error. Útil per trobar el problema que afecta als processos ràpidament.
- **Taula amb missatges informatius:** Taula que conté les dades de flux de les execucions afegides pel desenvolupador del procés. Útil per interpretar el que ha succeït durant una execució i trobar indicadors per solucionar errors.

Un cop esmentats els detalls de la implementació d'aquest dashboard, en la figura 10 es pot veure el resultat del desenvolupament d'aquest dashboard.



Fig. 10. Dashboard basat en Kibana.

9 TESTS

Un cop desenvolupada cada part s'ha sotmès a la fase de test on es comprova si l'aplicació actua com s'espera i satisfà els requeriments plantejats. Finalment també s'han realitzat tests d'acceptació on els usuaris han donat feedback de l'aplicació després d'haver-la utilitzat sota supervisió.

9.1 Dashboard Inicial

En el dashboard inicial els test han consistit a provar individualment cada element del sistema i fer proves específiques amb cadascun.

Els **Backend** s'han realitzat proves de caixa blanca on s'ha posat a prova tots els fluxes possibles dels backend mitjançant l'aplicació WCF Test client que ens permet testar els serveis web generats a partir de cada backend individualment introduint els camps d'entrada i obtenint els camps de sortida. Segons el tipus de dada i la interacció que té amb el sistema s'han introduït diferents entrades:

Int: -Valors < 1
-Valors >= 1
-Entrada null.

String: -Entrada null.
-Strings superiors a 255 caràcters.
-Strings inferiors a 255 caràcters.

Amb una bateria de proves que consisteix en introduir entrades totes les possibles combinacions de dades possibles en el WCF Test Client. S'ha comprovat que en cap dels casos s'ha generat un error d'execució i els fluxes utilitzats pels Backend són els esperats.

En la figura 11 es pot veure una de les proves.

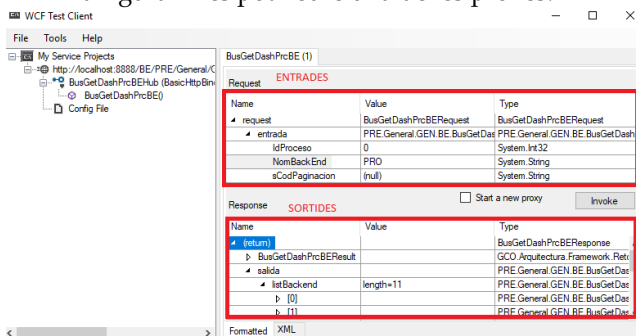


Fig. 11. Prova en WCF Test Client.

Un cop testejats els backend s'ha passat a fer-ho amb els **Frontend** que utilitzaran aquests frontend per mostrar a l'aplicació web les dades desitjades. Per realitzar aquest testeig s'ha utilitzat l'entorn de desenvolupament de l'empresa en el que es pot navegar en l'aplicació de forma natural i s'han comprovat que tots els possibles fluxos d'execució generen els resultats esperats. En aquesta fase s'han fet proves creuades utilitzant persones de l'entorn de treball com a testers que consistien en proposar funcions a realitzar dins l'aplicació i comprovar que podien fer-les correctament i cap element del sistema fallava. Algunes de les diferents tasques a realitzar pels testers són:

- Afegir un procés amb dos backends associats al sistema.
- Cercar un procés i modificar tots els paràmetres.
- Esborrar un procés del sistema.

Observant els testers i escoltant el seu feedback mitjançant aquestes proves s'ha comprovat que l'aplicació funciona correctament.

En quant al testeig al **model de dades**, s'ha realitzat una simulació en l'entorn de desenvolupament de la càrrega esperada en l'entorn de producció i s'ha mesurat la resposta de totes les consultes individualment. Això inclou la inserció d'aproximadament 90 processos al sistema amb els seus respectius Backends i una simulació d'un comportament real on s'utilitza l'aplicació per cercar i analitzar processos on s'ha comprovat que les consultes realitzades no suposen una baixada de rendiment en els servidors i responen ràpidament. Els temps de resposta de les consultes són inferiors a 1 segon així que es pot assegurar que la base de dades no rebrà una càrrega de treball problemàtica. També s'ha realitzat testejos comprovant que el model relacional està correctament implementat comprovant que a l'esborrar un element s'eliminen els elements referenciats per aquest.

9.2 Dashboard basat en Kibana

Aquest dashboard s'ha testejat inicialment comprovant que les dades dels logs s'han introduït correctament als servidors elastic search correctament i finalment realitzant tests d'integritat en l'aplicació Kibana.

Per comprovar si les dades dels logs han estat introduïdes correctament el testeig ha consistit en comprovar a través de Kibana si les dades introduïdes a Elastic Search es plasmaven correctament a l'aplicació Kibana.

La part gràfica del dashboard ha requerit un testeig d'integritat on s'han comparat exhaustivament que les dades representades a Kibana són les que es representen amb el log.

9.3 Tests d'acceptació

Finalment s'han realitzat tests d'acceptació entre els futurs usuaris, que són els cinc empleats encarregats de controlar els processos de l'aplicació, on han donat un feedback indicant dades el seu grau de satisfacció.

El test d'acceptació ha consistit en un qüestionari amb format Likert[8] on es proposen afirmacions respecte a l'aplicació com: "L'aplicació m'ha semblat intuïtiva." o "L'eina

em facilita la tasca de control de processos.”, on les respostes consisteixen a valorar com de d’acord està l’usuari amb les afirmacions amb una escala Likert de 5 punts amb les següents respostes:

1. Totalment en desacord
2. En desacord
3. Neutral
4. D’acord
5. Totalment d’acord

Aquestes respostes es punten de l’1 al 5, mesurant el grau de satisfacció de l’usuari. Aquestes puntuacions s’han mesurat, s’ha obtingut una mitjana i com a resultat a donat un 4.5. Aquest resultat s’ha interpretat com que l’aplicació està oferint el servei esperat i els usuaris estan satisfets amb el seu rendiment.

10 RESULTATS

Tot i que l’aplicació no s’ha arribat a implementar en l’entorn de treball objectiu encara, la part de l’aplicació que consisteix en el monitoratge a temps real sí que està sent productiva actualment i esta donant resultats satisfactoris a tots els membres de l’equip de Prepersa encarregats de controlar l’estat dels processos.

Els beneficis demostrats al monitorar processos són:

- Reducció d’errors diaris, ja que ara es poden prevenir errors amb més facilitat.
- Redueix el temps de cerca de processos en els servidors que els allotgen.
- Evita cerques de les descripcions de cada procés en el drive de documentació de processos.
- Reducció de càrrega d’emails de notificacions d’error als processos.
- Optimització en la programació temporal de les execucions de processos.
- Reducció en el temps de tractament dels errors.

11 PLANIFICACIÓ

La planificació s’ha hagut d’anar modificant durant el transcurs del projecte, ja que hi ha hagut diversos canvis en els requeriments de l’aplicació i s’ha hagut de reajustar. Inicialment s’inclouia més temps en el desenvolupament del projecte i finalment s’ha hagut de dedicar grans quantitats de temps en l’estudi i anàlisi de les possibles propostes.

Algunes de les tasques incloses a la planificació com millorar el contingut dels processos i incloure-hi més informació no s’ha pogut realitzar a causa d’incompatibilitats amb el sistema de l’empresa i s’han descartat.

A causa dels canvis de requeriments també s’ha accelerat el desenvolupament dels backend i frontend conjuntament amb les proves, per arribar a les deadline marcades.

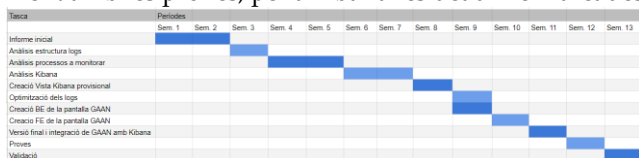


Fig. 12. Diagrama de Gantt.

12 CONCLUSIONS

En aquest document s’ha tractat el desenvolupament del projecte per resoldre les necessitats de l’empresa Prepersa de mantenir un control a temps real i continu de l’estat dels seus processos, així com facilitar el seu tractament i accés a l’equip de treball.

Inicialment s’ha realitzat un estudi sobre el sistema de l’empresa i l’estructura dels processos que ha permès dissenyar diferents propostes de solució per a ser avaluades pels clients. Mitjançant reunions amb els clients s’ha establert una proposta com la designada per desenvolupar i s’ha passat portat a la fase de desenvolupament.

S’han utilitzat diverses tecnologies ja donades per l’empresa que s’esmenten al document com Kibana o el framework GAAN, encarregat de la creació dels Frontend. A més l’estructura dels logs dels processos ja existia prèviament i s’ha utilitzat com a font de dades per analitzar l’estat dels processos.

En la fase de disseny s’ha mostrat el procés de disseny del model de dades necessari per crear el dashboard inicial. També es mostra com s’ha utilitzat un prototip per presentar la proposta de forma més clara als clients i facilitar el posterior desenvolupament de l’aplicació. Finalment s’ha mostrat un diagrama de casos d’ús, necessari per poder identificar totes les funcionalitats necessàries de l’aplicació i tenir-les en compte durant el desenvolupament.

En la fase de desenvolupament cal destacar que s’han realitzat dos dashboard diferents, un dashboard que proveeix informació estàtica útil des del qual s’accedeix al segon dashboard orientat a donar informació a temps real dels processos de l’empresa.

Després de la fase de desenvolupament s’expliquen els tests que s’han realitzat per comprovar que cada element desenvolupat compleix les funcions a les quals s’han designat conjuntament amb uns tests d’acceptació de l’aplicació que ajuden a validar la resolució del problema proposat.

Finalment s’analitzen els resultats obtinguts de la solució, observant els beneficis que aporta als clients i aportant informació del feedback d’aquests.

AGRAÏMENTS

M’agradaria agrair a Jose Lorente i Andrei Hernández, principals interessats en el projecte, i a l’equip del departament d’IT de Prepersa que m’han ofert la possibilitat de realitzar aquest projecte en el seu entorn i han aportat el seu granet de sorra. També vull agrair a Jonatan Hospital que ha col·laborat amb una formació sobre Kibana. Finalment he d’agradir també a Víctor Garcia Font pel suport i guia donat com a tutor durant el transcurs d’aquest projecte.

REFERÈNCIES

- [1] Prepersa, “Conocenos”, disponible en <https://www.prepersa.es/conocenos>. 2020.
- [2] Lladós Canet, J. “Diapositives d’Enginyeria del Software”. Scrum. 2018.
- [3] Kibana, “Kibana: Explora, visualiza y descubre datos | Elastic”, disponible en: <https://www.elastic.co/es/kibana>. 2020.
- [4] Logstash, “Logstash Introduction | Logstash Reference [7.6] | Elastic”, disponible en: <https://www.elastic.co/guide/en/logstash/current/introduction.html>.
- [5] Sai Divya, M. and Kumar Goyal, S., 2013. Elasticsearch An Advanced And Quick Search Technique To Handle Voluminous Data.
- [6] Elastic search, “Ingest node | Elasticsearch Reference [7.6] | Elastic”, disponible en: <https://www.elastic.co/guide/en/elasticsearch/reference/current/ingest.html>. 2020.
- [7] Grok Patterns, “Kibana: Explora, visualiza y descubre datos | Elastic”, disponible en: https://streamsets.com/documentation/datacollector/latest/help/datacollector/UserGuide/Apx-GrokPatterns/GrokPatterns_title.html. 2020.
- [8] Oriol Llauradó, “La escala de Likert: qué es y cómo utilizarla”, disponible en: <https://www.netquest.com/blog/es/la-escala-de-likert-que-es-y-como-utilizarla>. 2020.